

**REMARKS**

Claims 1-32 are pending in this application

**Rejections under 35 U.S.C. § 103**

Claims 1-20, 28, 29, and 32 were rejected under 35 U.S.C. 103(a) as being unpatentable over "Clustering: Transparent Replication, Load Balancing, and Failover" by Salil Deshpande, published January 2000 (hereinafter "Deshpande reference"). Applicants respectfully traverse this rejection. In addition, the Office's comments in the Final Office Action supporting the rejections are traversed. As will be fully explained, the Deshpande reference does not disclose or suggest each and every feature of independent claims 1, 11, and 16 as required to raise a prima facie case of section 103 obviousness against independent claims 1, 11, and 16. Applicants incorporate by reference the arguments regarding the section 103 rejection in the remarks section of the Amendment filed on May 12, 2004.

Applicants respectfully submit that a prima facie case of obviousness must include the disclosure or suggestion of every feature of the claimed invention. Applicants submit that each and every feature of the claimed invention is not disclosed or suggested by the cited prior art references. The Office has explicitly stated that Deshpande does not disclose the replicated state manager but states that "it would have been obvious to one skilled in the art of the invention that Deshpande would have had some form of replicated state manager" and cites to Deshpande's failover approach on page 14 as support for this suggestion. Applicants respectfully submit that by this statement, the Office is utilizing hindsight. Basically, it appears that even though Deshpande does not disclose or suggest the invention, the Office is imputing the replicated state manager to the teachings of Deshpande even though Deshpande itself does not discuss or suggest utilization of a replicated state manager. Applicants respectfully submit that even if it is possible for Deshpande to use a replicated state manager

as claimed in the present application, that does not mean that Deshpande is actually using a replicated state manager.

Moreover, Applicants respectfully submit that some suggestion or motivation must be in the prior art to modify the teachings of Deshpande to generate the claimed invention. The Office appears to suggest that one skilled in the art would have made the modifications to Deshpande without showing that such a modification was suggested or motivated by the prior art. In addition, Applicants respectfully submit that the prior art must suggest the desirability of the claimed invention. Once again, Applicants submit that just because a modification of the teachings of Deshpande is possible is not sufficient to establish a prima facie case of obviousness. Applicants respectfully submit that some of the arguments below show that the Deshpande reference does not suggest the desirability of the claimed invention.

Applicants respectfully submit that Deshpande does not suggest a replicated state manager that can use an in-memory database. The Office uses the Deshpande for the suggestion that in-memory database is generally used in the replication of servlet session states and implies that this supports the Office's suggestion that an in-memory database for the replicated state manager is utilized. The Office further suggests that "Deshpande's statement that an in-memory database may not be suitable does not teach away from the use of an in-memory database he states that it is still generally used despite this flaw." Furthermore, the Office appears to suggest the combining of Deshpande's teachings regarding servlet session state replication (page 10 of Deshpande) with Deshpande's fail over approach in the Borland AppServer (page 14 of Deshpande). Applicants respectfully traverse the Office's suggestions.

The Office as indicated above suggests that the replicated state manager would be obvious to one skilled in the art due to Deshpande's failover teachings on page 14. (See page 2 of Final Office Action) and the Office further suggests that the in-memory database would

be obvious to one skilled in the art due to the Deshpande's servlet session state replication teachings on page 10 (See page 2 of Final Office Action). The Office appears to suggest that the combination of these teachings could be utilized to generate the feature of a replicated state manager that uses an in-memory database. Applicants traverse these suggestions.

First, Deshpande considers use of in-memory storage to store servlet session states as a flaw. Applicants respectfully submit that Deshpande goes on to recommend against usage of the servlet session state replication. The Office is respectfully directed to section 6.5.3. on page 10 of Deshpande which states in pertinent part:

Generally, the replication for such session state is in-memory, and not disk- or database-based. It is thus still susceptible to failures, and not suitable for persistent components.

The above quote discusses the use of servlet session state replication. In the same section as the above cited portion (i.e., section 6.5.3. on page 10 of the Deshpande reference), the Deshpande reference states the following:

***Therefore, we do not recommend the use of such Servlet session state replication.*** The Servlet engine of Borland AppServer does not perform such Servlet session state replication. As described later in this paper, BAS instead concentrates on providing solid load balancing, replication and failover support at the EJB container level, a more superior approach. (Emphasis in Original)

As can be seen, Deshpande recommends against the usage of servlet session state replication using in-memory storage with the failover approach of the Borland AppServer. Therefore, Applicants respectfully submit that the Office is suggesting a combination of teachings which Deshpande does not recommend. As a result, Applicants submit that even if the teachings of Deshpande could be combined to generate the claimed invention (a proposition with which Applicants disagree), one skilled in the art would not be motivated to use an a replicated state

manager with an in-memory database when the recommendations of Deshpande are considered.

Additionally, Applicants submit that Deshpande does not discuss usage of both an in-memory database and a replicated state server which can both contain entity bean states that can be recovered. To this statement, the Office states on pages 2 and 3 of the Final Office Action that:

"teaching of a Servlet session, on page 10, provides a system that stores replicas of the system state across multiple nodes, and this information is generally stored in an in-memory database. These multiple nodes act as replicated state servers, all of which can use the in-memory databases to recovery entity bean states, which are part of the system state."

Applicants traverse this suggestion and respectfully submit that the Office attempts to combine the servlet session state replication teachings of Deshpande on page 10 with the fail over teachings of Deshpande on page 14 to support its suggestion that the usage of both an in-memory database and a replicated state server which can both contain entity bean states that can be recovered would be obvious to one skilled in the art. As discussed above, Applicants respectfully submit that Deshpande does not recommend such a suggestion. Consequently, Applicants respectfully submits that even if the teachings of Deshpande on pages 10 and 14 could be combined to generate the claimed invention (a proposition with which Applicants disagree) one skilled in the art would not have been motivated to combine the teachings of Deshpande on page 10 (servlet session state replication) and page 14 (BAS implementation) to generate the claimed invention as is suggested by the Office.

Claim 11 includes the feature of the first state object storing a state of the entity bean object within a memory address space of a J2EE server process. As discussed above with respect to claim 1, Deshpande does not disclose or suggest the storing of the state of the entity bean object within a memory address space.

Claim 11 also includes the feature of using a replicated state server. As a result, Applicants also respectfully submit that as discussed above, Deshpande does not disclose or suggest the usage of in-memory storage in addition to the replicated state server. Consequently, Applicants respectfully submit that one skilled in the art would not have used the teachings of Deshpande to generate the claimed invention of claim 11 for at least the same reasons as discussed for claim 1.

Claim 16 includes the features of a managed state part having a first state object storing a state of the entity bean object and the second state object storing a state of the related entity bean object. Applicants respectfully submit that Deshpande neither discloses nor suggests storing a state of an entity bean within a state object of a managed state part. In the Final Office Action, the Office cites to section 8.4 on page 15 of Deshpande to support its suggestions that the above-described features are obvious. Applicants respectfully submit that section 8.4 of Deshpande discusses a situation where replicas of entity beans E1 and E2 are located on different machines. In addition, Applicants submit that the term "replicas" as used by Deshpande appears to be discussing an "instance of". Therefore, Applicants respectfully submit that section 8.4 does not teach the managed state part of claim 16.

Continuing with the discussion about claim 16, with respect to the Office's suggestion that the use of two entity beans in the same application session is the relationship of sharing the same session, Applicants respectfully traverse this suggestion and respectfully submit that just because two entity beans are used in a same application session does not necessarily show maintaining of a relationship. Applicants respectfully submit that claim 16 includes the feature where "the managed state part maintains a relationship between the entity bean object and the related entity bean object". Applicants respectfully submit that Deshpande does not disclose or suggest this feature.

Claim 16 further includes the feature of a state object that stores a state of an entity bean and also includes the feature of a replicated state server that stores a replica of the state object. Deshpande generally teaches the fail over from entity beans in one container to entity beans of the same type in a different container. Therefore, Deshpande teaches having one copy of a past state of the entity bean. The Office cites section 8.2 of Deshpande to support its suggestion that the above-described feature in claim 16 is obvious. Applicants respectfully traverse this suggestion. Applicants respectfully submit that section 8.2 of Deshpande discusses usage of one copy of the entity bean state in a different container. With respect to claim 16, Applicants submit that the first state object stores a state of the entity bean object in the application part and the replicated state server stores a replica of the first state object. Consequently, Applicants respectfully submit that Deshpande does not disclose or suggest all of the features of claim 16.

Applicants therefore respectfully submit that Deshpande does not disclose or suggest all of the features of the claimed inventions. Applicants respectfully submit that the dependent claims are allowable for at least the same reasons as the independent claims. Consequently, Applicants respectfully request that the section 103 rejections be withdrawn.

**Allowable Subject Matter**

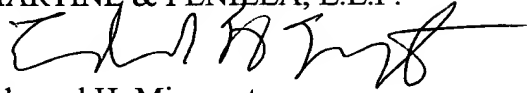
The Examiner has stated that claims 21-27, 30, and 31 would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims. Applicants respectfully submit that, for the reasons discussed above, claim 16 is allowable and therefore claims 21-27, 30, and 31 are allowable in their present form.

In view of the foregoing, Applicants submit that these claims are in condition for allowance. Accordingly, a notice of allowance is respectfully requested. In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 749-6900 ext. 6927. If any additional fees are due in

Application No. 09/818,214  
Response to Final Office Action dated September 24, 2004  
Correspondence in response to the Final Office Action mailed June 24, 2004

connection with the filing of this paper, then the Commissioner is authorized to charge such fees to Deposit Account No. 50-0805 (Order No. SUNMP005).

Respectfully submitted,  
MARTINE & PENILLA, L.L.P.



Edmund H. Mizumoto  
Reg. No. 46,938

710 Lakeway Drive, Suite 170  
Sunnyvale, California 94085  
(408) 749-6900  
Customer Number 32291